Performance Evaluation of Two-Space Genetic Algorithm for Optimizing Load on Video on Demand Servers

Aruna Yadav, Saurav Chandra, Sanjeev Kumar
Krishna Institute of Engineering & Technology, Ghaziabad, India

ABSTRACT

To improve the performance of video on demand servers there is need of selecting an appropriate load balancing technique so requests can be distributed in optimized manner .To meet the growth of web traffic the concept of load balancer was introduced. The role of load balancer is to distribute the tasks among the web servers efficiently. In this paper system compared algorithms used for distributing the loads are: FCFS, Genetic, and two space GA algorithms. Performance of algorithms is calculated on the basis of makespan and average resource utilization. Two- Space Genetic Algorithm is proved better over other server selection techniques. Two-Space Genetic Algorithm gave lower makespan and higher resource utilization.

General Terms

Soft Computing and Optimization

Keywords

Video-on-demand Server, Make span, CPU-Utilization, Fitness Function, FCFS, Genetic Algorithm, Two-Space-Genetic Algorithm.

1. INTRODUCTION

Current trends of internet usages had shown dramatically demand of video streams and overloading of servers. Multimedia Mail, Digital Libraries, Video Conference (VC), IP telephony are some most popular applications [1, 2]. There is more and more demand of video content on web so to meet such demand system require multiple web servers or server farms. Web server is a program that provides content like web pages over the World Wide Web. On the requests from clients, these web servers' uses own operating system may be of different types. Large numbers of requests come from user, so they have to balance load of requests to improve performance of web servers.

It is considered a specific application of web servers: Video on Demand. Video on Demand servers must be consistent to transfer higher data rate [2,8]. Use of on demand video services have amplified significantly in the recent years and affected the performance of web servers and is predictable to rise further due to expansion in technology . So challenge is to meet the high quality of services required by Video on Demand applications. Although when user demands and user access rates increase some problems are faced like high block rate, long startup delay, service interruption and frame losing. The qualities of services as recognized by the users are generally dependent on the requirements. So they must be according to the requirements and traced to an appropriate objective so that it will be technically correct required application.

The simultaneous request from users produce load on web servers. Thus the waiting time becomes high when the numbers of requests to the web server increases, resulting in Denial of Service attack .To solve this problem multiple servers are used known as clustered Web Servers or a server farm.

To serve a large number of requests server replication takes extra charges for installation of new servers up to a limit and beyond a certain number of servers, further increase will only take more installation charges without improving the quality of services like throughput, speedup, waiting time etc.

In today scenario Multimedia communications is used in large extent .In multimedia communication requests are received and transfer the information accordingly. These services should respond continuously and with minimum delay, server farm helps it to make better. The performance of a server farm can improve with the improvement of type of routing method, server capacity and scheduling policies used.

The server capacity is of two types homogeneous or heterogeneous. Heterogeneous systems give better results than homogeneous systems if tasks are of different sizes. Heterogeneous systems can also include task-specific systems.

These servers should have individual operating systems and may provide load balancing approach. Load balancing on servers plays vital role to improve the performance when there are lot of server requests. In loads balancing policy, policy focus on task location policy which describes scheduling algorithm for various tasks. Scheduling algorithms are described by policy through which they allocate tasks to different web servers. Here problem will go through the scheduling algorithms i.e. first come first serve, genetic algorithm and Two-Space Genetic Algorithm.

Makespan is a performance parameter through which system compare different scheduling algorithms. Maximum time consumed to complete all the tasks in the bar given to the dispatcher or load balancer is called Makespan. There are some other performance parameter like average-cpu-utilization, fitness function and cost [2,9].

2. LITERATURE SURVEY

In vector evaluated genetic algorithm, VEGA each sub population is evaluated with respect to different objective and straight forward implementation but this algorithm converge to the extreme of each objective [13].

MOGA, multi objective genetic algorithm based on pareto ranking, simple extension of single objective Genetic Algorithm but its convergence is too slow [14].

A weight based genetic algorithm, WBGA takes weighted average of normalized objectives. It is simple extension of single objective Genetic Algorithm but there are difficulties in non-convex objective function space [15].

DMOEA, Dynamic Multi Objective Evolutionary Algorithm do cell based ranking but its implementation is difficult [16].

In Rank-density- Based Multi-Objective Genetic Algorithm, RDGA reduces problem to bi-objective problem but its implementation is difficult [17].

In Strength Pareto Evolutionary Algorithm, SPEA ranking based on the external archive of non-dominated solutions but there are complex clustering algorithms [18].

SPEA-2 has strength of dominators .It is better than SPEA but computationally expensive fitness and density calculation. 19].

In Non-dominated Sorting in Genetic Algorithms , NSGA ranking based on non-domination sorting. It has fast convergence [20].

NSGA-2 single parameter well tested efficiently but crowding distance works in objective space only [21].

In Niched Pareto Genetic Algorithm, NPGA has no fitness assignment. It has very simple selection process with tournament selection but there are extra parameters for tournament selection [22].

A Random Weighted Genetic Algorithm, RWGA takes weighted average of normalized objectives. It is easy to implement but there are difficulties in non-convex objective function space [23].

PESA, Pareto Envelope-Based Selection Algorithm has no fitness assignment. It is easy to implement and computationally efficient but performance depends on cell sizes [24].

In Pareto Archived Evolution Strategy, PAES uses pareto dominance to replace a parent if offspring dominates but this is not a population based approach [25].

Shopova [3] showed that GA used real representation schemes for both real and integer variables. In representation methods are dynamic representation, real representation of integer variables. In selection for reproduction methods are Roulette Wheel Selection, Rank based selection and Tournament selection. In Crossover methods are single point crossover, double point crossover, N-point-crossover, uniform crossover, arithmetical crossover, blend crossover. In mutation schemes are uniform mutation, non-uniform mutation and breeder mutation.

Chande and Sinha [4] explained Genetic Algorithm is a versatile optimization tool. Genetic Algorithm is invented by John Holland, it is an abstraction of biological evolution. Algorithm works by performing representation, evaluation, selection, recombination, mutation and inversion. They quoted some application of GA in real world : Nutrional Counseling , Stylometry, Parametric Design of a aircraft, Robot trajectory generation, Strategy acquisition for simulated airplanes, Redistricting, Problem solving and in-circuit emulators , Acoustics , Aerospace engineering , Bandwidth optimization in near video on demand system , Medical , Scheduling , Musical Composition , Finance , Identifying criminal suspect , Seeking Routes.

Selecting a server and optimized load balancing is studied from various research papers. Some server selection algorithms [5, 6] are the closest server algorithm that selects server based on the requirement of the client, optimized closest server algorithm that chooses the closest server among the free channels.

According to Bajpai and Kumar [7] Genetic Algorithm works as a global optimization approach.GA are intrinsically parallel and perform well in problem for which the fitness landscape is complex.

According to Gupta et al. [8] analyzed that minimum expected cost algorithm computes mainly server parameters like latency and bandwidth.

Niyato et al. [9] showed load balancing for Internet video and audio server and compared algorithms like round-robin, FCFS, max-min and random traditional algorithms along with Adaptive bidding, Diffusion and State change broadcast. Wang et al. studied load balancing in different servers with different service rates and then observed it for heterogeneous systems of multiple servers. This was implemented by heuristic methods by multiple thresholds setting.

Ciardo et al. [10] analyzed on a idea based on size distributions of the requested documents to allocate the tasks to web servers.

Zhang et al. [11] derived average response time and the rejection rate and compared three different routing policies to analyze the central load balancing model.

Jeffrey W. Herrmann [26] explained to take decision under uncertainty is a tough task although structuring decision under uncertainty leads to effective decision making. Two-Space Genetic Algorithm works well for large population and able to manage uncertainty.

For better results decisions are taken under uncertainty which is a difficult problem. Then, after solving the associated optimization problem, one can select the decision that has the best average performance over time or one can select the decision that has the best performance in the expected outcome. Robust discrete optimization, on the other hand, seeks to identify decisions that will perform well under any circumstances.

3. PROBLEM STATEMENT FOR VIDEO ON DEMAND SERVERS

If there are n number of task then system have to map all the task among m possible servers, so problem is to find best server selection algorithm to improve the performance metrics. There are many factors like waiting time,server load, response time, cpu utilization ,makespan that impact the quality of service. The server selection algorithms can be compared by different metrics and this depends on the task set where a particular metric or metrics are of more significance than others. For getting better results, system have to foucs on fitnes value.Fitness value depends upon makespan,average utilization and number of processors.Makespan is the largest completion time of all the tasks in the system. Fitness=(1/ makespan) x (average utilization) x (1/no. of processors).Those two strings have better fitness value select them and apply crossover and mutation operation. To get the optimized result perform crossover operation until fitness value reaches greater than 0.4.[27]

```
VOD_ALGO (A, N)
1. For i ←1 to n
2. Create_evpop (A)
3. Calculate makespan
4. Calculate average_utilization
5. Calculate fitness value
6. Go to step 2
```

7. Select two best fitness f1 and f2.

8. Crossover (f1, f2)

9. If probability of fitness value string < 0.4

10. Go to step 8

11. Mutation(x1, x2)

12. End [27]

4. PERFORMANCE EVALUATION OF TWO-SPACE GENETIC ALGORITHM

Now in this paper two-space genetic algorithm proposed by Jeffrey W. Herrmann in 1999 [26] is used for finding best possible solutions for optimizing selection of load on Video-On-Demand servers.

When started to work on this problem, first go through the steps of two-space genetic algorithm and then go for the details. Different thing is that the two-space genetic algorithm maintains two distinct populations: Let A1 has some strings that represent solutions in X, and A2 has strings that represent solutions in S. For a solution x in A1, the objective function h(x) evaluates that solution's worst-case performance with respect to the second population:

$h(x) = \max\{F(x, s) : s \in A2\}$

The algorithm is against large h(x) and goes with small h(x), so solutions with better worst-case performance will survive. Similarly, for a scenario s in A2, the objective function g(s) evaluates the best solution in the first population:

$g(s) = \min\{F(x, s) : x \in A1\}$

The algorithm is against small g(s) and goes with large g(s), so scenarios with worse optimal solutions will survive.

A simple genetic algorithm has the following steps:

1. Create initial generation

2. For each individual evaluate its fitness f(i).

3. Create next generation by reproduction, crossover, and mutation.

4. Continue it unless get desired results, Return to Step 2.

The two-space genetic algorithm can be summarized as follows [26]:

1. Create initial generations A1(0) and A2(0). Let t = 0.

2. For each individual $x \in A1(t)$, evaluate $h(x) = \max\{F(x, s) : s \in A2(t)\}$.

3. For each individual $s \in A2(t)$, evaluate $g(s) = \min\{F(x, s) : x \in A1(t)\}$.

4. Create generation A1 (t+1) by reproduction, crossover, and mutation.

5. Create generation A2 (t+1) by reproduction, crossover, and mutation.

6. Let t = t + 1. Unless t equals the maximum number of generations, return to Step 2.

The two-space genetic algorithm lessens the computational effort needed. It takes a sample population from the set of scenarios and allows this to progress while the algorithm is searching for solutions. Thus, it searches two spaces simultaneously. Moreover, it is evaluating the solutions in parallel, since it uses the same scenarios for all solutions.

5. RESULTS AND DISCUSSION

Two-Space-GA simultaneously works on two populations. Like

Population1: fitness is fit1=0.008700 and fit2=0.022789

Population 2: fitness is fit3=0.244892 and fit4=0.315591

For next iteration:-

Population 1: now fitness is fit1=0.013043 and fit2=0.018918

Population 2: now fitness is fit3=0.100469 and fit4=0.077910

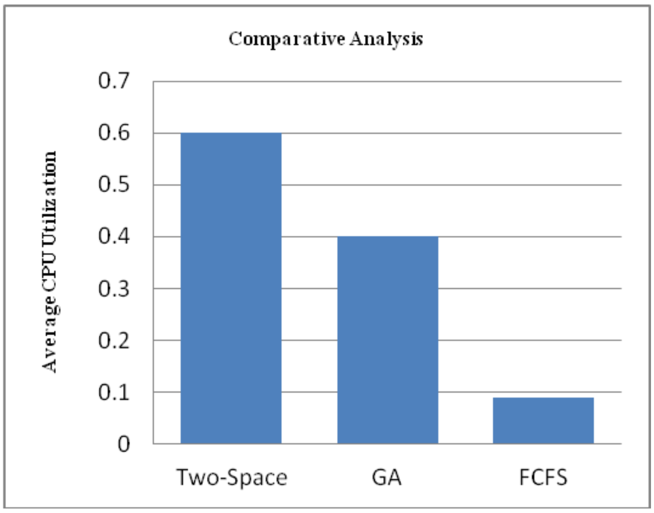

Fig 1: Comparison of CPU Utilization among Two-Space, GA and FCFS

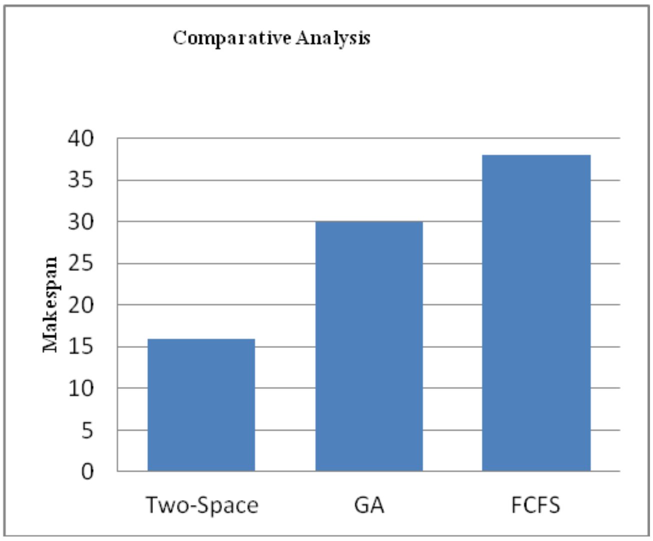

Fig 2: Comparison of Makespan among Two-Space, GA and FCFS

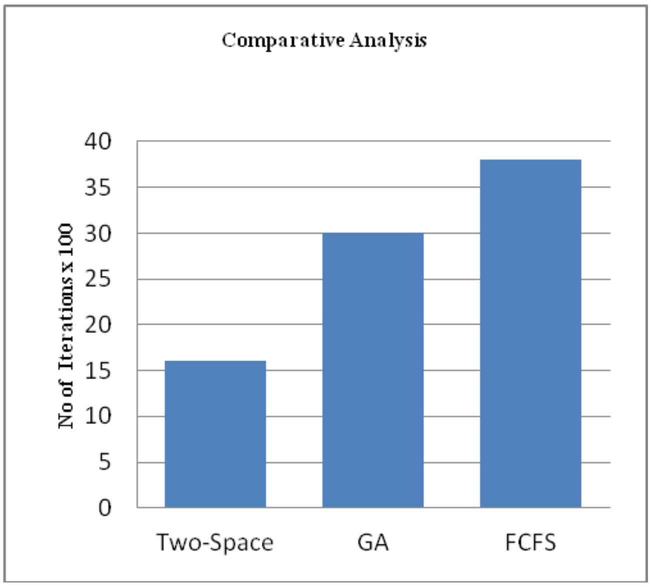

Fig 3: Comparison among Two-Space, GA and FCFS

6. CONCLUSION AND FUTURE SCOPE

Two-Space provides far better results than FCFS and GA .As it is very clear from the graphs that Two-Space-GA is efficient algorithm than GA and FCF. In different criteria Two-Space-GA proved itself that it provides better average cpu utilization and lowest makespan. To get more optimized results hybrid technology can be used.

To get more optimized results hybrid optimization techniques can be used, which may produce more efficient and optimized result than Two-Space –GA.

7. REFERENCES

[1] N. Panigrahi and B. Sahoo. may 08 2011 Qos Based Retrieval Strategy for Video on Demand.Available online at http://dspace.nitrkl.ac.in:8080/dspace/bitstream/2080/789/1/bdsahoo-2009.pdf.

[2] Das.Suraj, Prusty.Alok and Sahoo.Bibhudatta. Performance Analysis of Server Selection Schemes for Video on Demand servers in N.I.T.Rourkela.

[3] E.G. Shopova, N.G. Vaklieva-Bancheva. march 2006. Basic –A Genetic Algorithm for Engineering Problem Solution, Computers and Chemical Engineering,"Volume 30,

[4] Chande Swati V. and Sinha Madhavi 2008 Genetic Algorithm: A Versatile Optimization Tool, Bidyapeeth Internatonal Journal of Information Technology.

[5] Tsutsui, Shigeyoshi, and Ashish Ghosh, 1997 Genetic Algorithms with a Robust Solution Searching Scheme, IEEE Transactions on Evolutionary Computation, Volume 1,Number 3, pages 201-208.

[6] M. Ko and I. Koo. Dec 13, 1996 An Overview of Interactive Video On Demand System.Technical Report, The University of British Columbia.

[7] M. Guo et al. May 2002 Selecting among Replicated Batching Video on Demand Servers. Proceedings of the 12th international workshop on Network and operating systems support for digital audio and video.

[8] V. Gupta, M.H. Balter, K. Sigman and W.Whitt. October 2007.Analysis of Join-the-Shortest-Queue Routing for Web Server Farms.Performance Evaluation,vol. 64, no. 9-12, pp. 1062-1081.

[9] D. Niyato and C.Srinilta." Load Balancing Algorithms for Internet Video and Audio Server". Proceedings of 9th IEEE International Conference on Networks, pp. 76, 2001.

[10] G.Ciardo, A.Riska and E.Smirni. EQUILOAD:" A Load Balancing Policy for Clustered Web Servers". Performance Evaluation,vol. 46, no.2-3, pp. 101-124, October 2001.

[11] Z. Zhang and W.Fan. "Web Server Load Balancing: A Queueing Analysis.European Journal of Operational Research", vol. 186, no. 2, pp. 681-693, April 2008.

[12] Goldberg, D.E., "Genetic Algorithms in Search, Optimization, and Machine Learning", Addison-Wesley, Reading, Massachusetts, 1989.

[13] Schaffer JD. "Multiple Objective Optimizations with Vector Evaluated Genetic Algorithms". In: proceedings of the international conference on genetic algorithm and their applications, 1985.

[14] Fonseca CM, Fleming PJ." Multi-Objective Genetic Algorithms". In: IEE colloquium on 'Genetic Algorithms for Control Systems Engineering' (Digest No. 1993/130), 28 May 1993. London, UK: IEE; 1993.

[15] Hajela P, lin C-y. "Genetic Search Strategies in Multi-Criterion Optimal Design". Structural Optimization 1992;4(2):99–107.

[16] Yen GG, Lu H. Dynamic "Multi Objective Evolutionary Algorithm: A Adaptive Cell-based Rank and Density Estimation". IEEE Trans Evol Comput 2003; 7(3):253–74.

[17] Lu H, Yen GG. "Rank-density- Based Multi-Objective Genetic Algorithm and Benchmark Test Function Study". IEEE Trans Evol Comput 2003; 7(4):325–43.

[18] Zitzler E, Thiele L. "Multi Objective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach". IEEE Trans Evol Comput 1999;3(4):257–71.

[19] Zitzler E, Laumanns M, Thiele L. "SPEA2: Improving the Strength Pareto Evolutionary Algorithm". Swiss Federal Institute Technology: Zurich, Switzerland; 2001.

[20] Srinivas N, Deb K. "Multi Objective Optimization Using Nondominated Sorting in Genetic Algorithms". J Evol Comput 1994;2(3):221–48.

[21] Deb K, Agrawal S, Pratap A, Meyarivan T. "A fast Elitist Nondominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II". In: Proceedings of sixth international conference on parallel problem solving from nature, 18–20 September, 2000. Paris, France: Springer; 2000.

[22] Horn J, Nafpliotis N, Goldberg DE. "A Niched Pareto Genetic Algorithm for Multi-Objective Optimization". In: Proceedings of the first IEEE conference on evolutionary computation. IEEE World Congress on Computational Intelligence, 27–29 June, 1994. Orlando, FL, USA: IEEE; 1994.

[23] Murata T, Ishibuchi H. MOGA:" Multi-Objective Genetic Algorithms". In: Proceedings of the 1995 IEEE

international conference on evolutionary computation, 29 November–1 December, 1995. Perth,WA, Australia: IEEE; 1995.

[24] Corne DW, Knowles JD, Oates MJ. "The Pareto Envelope-Based Selection Algorithm for Multi-Objective Optimization". In: Proceedings of sixth international conference on parallel problem solving from Nature, 18–20 September, 2000. Paris, France: Springer; 2000.

[25] Knowles J, Corne D." The Pareto Archived Evolution Strategy: A New Baseline Algorithm for Pareto Multi-Objective Optimization". In: Proceedings of the 1999 congress on evolutionary computation-CEC99, 6–9 July 1999. Washington, DC, USA: IEEE; 1999.

[26] Herrmann, Jeffrey W "A Genetic Algorithm for Minimax Optimization Problems" Proceedings of the 1999 Congress on. Evolutionary Computation, 1999.

[27] Yadav Aruna,Kumar Sanjeev"Genetic Algorithm for Optimizing Load Distribution on Video on Demand Servers"International Journal of Scientific & Engineering Research Volume 3,Issue 3,March-2012.